REMARKS/ARGUMENTS

Claims 1-6, 8-16, and 18-20 are pending in the present application. Claims 1, 10, 11, 19, and 20 are amended in the present response. Claims 21 and 22 have been added. Reconsideration of the claims is respectfully requested.

I. Examiner Interview

Applicants would like to thank the Examiner for conducting an interview on October 12, 2007. The rejections under § 103 were discussed. No agreement was reached.

II. 35 U.S.C. § 103, Obviousness

The Examiner rejects claims 1-6, 8-16, and 18-20 under 35 U.S.C. § 103 as obvious over *Carlson*, System and Method for Caching Javaserver Pages™ Responses, U.S. Patent No. 6,697,849 (February 24, 2004) (hereinafter "*Carlson*") in view of in view of *Jindal* et al., Load Balancing in a Network Environment, U.S. Patent No. 6,327,622 (December 4, 2001) (hereinafter "*Jindal*"). This rejection is respectfully traversed. In regards to claim 1, the Office Action states the following:

> Referring to claim 1, Carlson discloses a method of distributing traffic to application instances (i.e. applications 202-208 running on application server 200) on one or more computing devices (i.e. servers 308A-C), comprising:
>
> obtaining application instance specific operational information (i.e. server load criteria and application component performance criteria) identifying operational characteristics (i.e. elements shown in Figures 8 and 9) of an application instance on a computing device on the one or more computing devices (e.g. abstract; col. 12, lines 40-67);
>
> generating a load balancing weight to be associated with an application instance based on the application instance specific operational information (i.e. random number is generated in a weighted manner according to the "best" server at that particular time) (col. 16, lines 13-47); and
>
> distributing traffic based on the generated load balancing weight (i.e. "gracefully" distribute requests among the application servers) (col. 16, lines 35-47).
>
> Carlson does not explicitly disclose that the instance specific operational information includes a number of successful transactions processed by the application instance in a given period of time. In analogous art, Jindal discloses another method for distributing traffic to application instances which discloses utilizing throughput in the criteria for load balancing (col. 2, lines 65-67). It would have been obvious to one of ordinary skill in the art to combine the teaching of Jindal with Carlson in order to utilize the policies of Jindal with the performance criteria used by Carlson, thereby increasing the ability to customize load balancing weights according to the user's liking.

Office Action dated July 13, 2007, pages 2-3.

The Examiner bears the burden of establishing a *prima facie* case of obviousness based on the prior art when rejecting claims under 35 U.S.C. §103. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992). Additionally, all limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Therefore, no *prima facie* obviousness rejection can be established if the proposed combination does not teach all of the features of the claimed invention. Furthermore, if an independent claim is nonobvious under 35 U.S.C. 103, then any claim depending therefrom is nonobvious. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988).

Claim 1, as amended, is as follows:

> 1. (Currently Amended) A method, in a data processing system, of distributing traffic to application instances on one or more computing devices, comprising:
>
> obtaining application instance specific operational information identifying operational characteristics of an application instance on a computing device of the one or more computing devices, wherein the application instance specific operational information includes at least one of ~~a number of successful transactions processed by the application instance within a period of time,~~ an application instance topology, an importance of transactions currently being processed by the application instance, an amount of time the application instance has been blocked waiting for resources, and an amount of resources consumed by the application instance;
>
> generating a load balancing weight to be associated with the application instance based on the application instance specific operational information obtained; and
>
> distributing the traffic to the application instance based on the load balancing weight.

No *prima facie* obviousness rejection can be stated because neither reference, alone or in combination, teaches all of the features of amended claim 1. *Carlson* does not teach all of the features claim 1 as amended, because *Carlson* does not teach the feature "obtaining application instance specific operational information identifying operational characteristics of an application instance on a computing device of the one or more computing devices, wherein the application instance specific operational information includes at least one of *an application instance topology, an importance of transactions currently being processed by the application instance, an amount of time the application instance has been blocked waiting for resources, and an amount of resources consumed by the application instance.*" The Examiner does not assert otherwise.

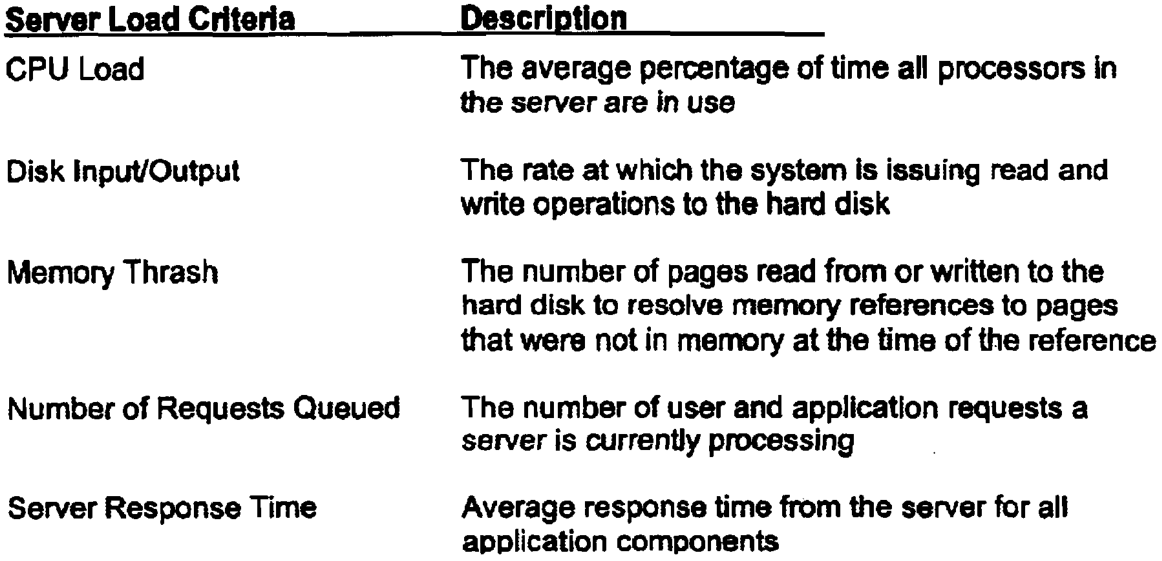
The Office Action states that *Jindal* discloses another method for distributing traffic to application instances which discloses utilizing throughput in the criteria for load balancing. Claim 1 as amended excludes the limitation "a number of successful transactions processed by the application instance within a period of time" as previously presented. *Jindal* does not teach the remaining limitations of the amended feature of claim 1. The Examiner does not assert otherwise. Thus, the combination of *Carlson* and *Jindal* does not teach all of the features of amended claim.

Independent claims 11 and 20 recite similar amendments as presented in claim 1. Additionally, Applicants have amended claims 10 and 19 to exclude the limitation "which of the application instance and the one or more other application instances has a relatively better response time." *Carlson* does not teach the remaining limitations of claims 10 and 19. The remaining claims depend from claims 1 and 11, respectively; therefore, the same distinctions between the cited references vis-à-vis claim 1 apply to the respective dependent claims. Therefore, the rejection of claims 1, 4-8, 10-11, and 14-20 under 35 U.S.C. § 103 has been overcome.

Moreover, the proposed combination of *Carlson* and *Jindal* when considered as a whole does not teach or suggest all of the features of the new claims. For example, neither *Carlson* nor *Jindal* teach the recited feature of claim 21. Claim 21 is reproduced below:

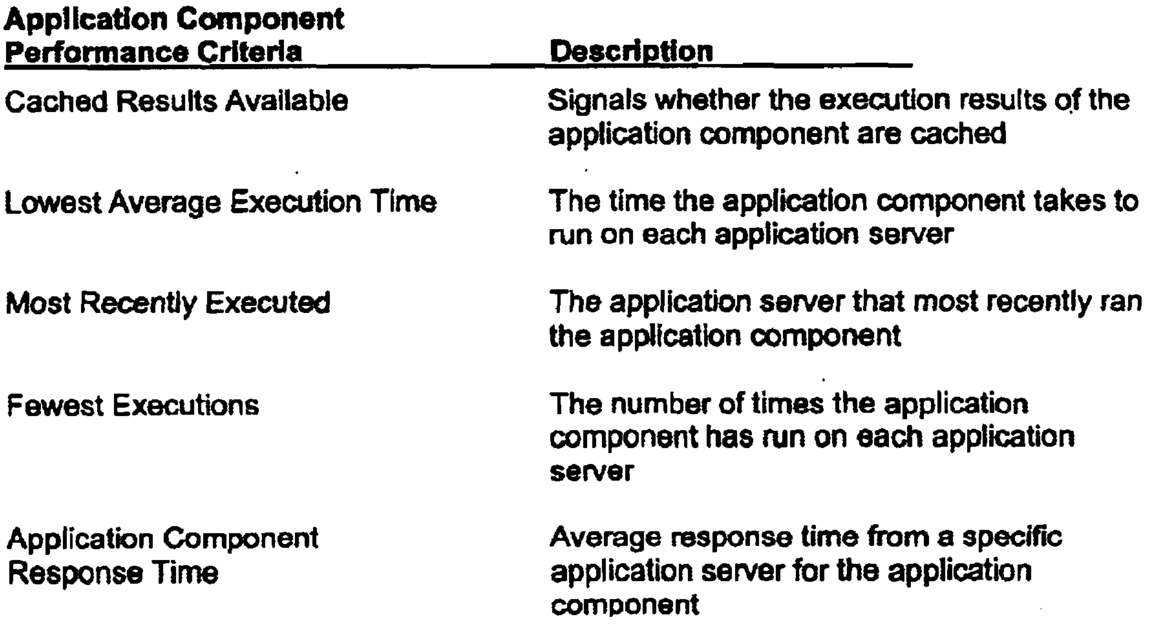
> 21. (New) The method of claim 4, wherein generating the load balancing weight based on the relationship between the application instance specific operational information and the one of more other application instance specific information includes:
>
> assigning a base weight to each of the application instance and the one or more other application instances; and
>
> increasing a weight value associated with the application instance or the one or more other application instances based on which of the application instance and the one or more other application instances processes the least significant transactions.

Neither *Carlson* nor *Jindal* teach "increasing a weight value associated with the application instance or the one or more other application instances based on which of the application instance and the one or more other application instances processes the least significant transactions" as recited in claim 21. *Carlson* teaches a method for caching JavaServer Page™ (JSP) component responses. *Carlson*'s figures 8 and 9 depict the server load criteria and application component performance criteria used for load balancing. Figures 8 and 9 are reproduced below:



Server Load Criteria	Description
CPU Load	The average percentage of time all processors in the server are in use
Disk Input/Output	The rate at which the system is issuing read and write operations to the hard disk
Memory Thrash	The number of pages read from or written to the hard disk to resolve memory references to pages that were not in memory at the time of the reference
Number of Requests Queued	The number of user and application requests a server is currently processing
Server Response Time	Average response time from the server for all application components

Fig. 8



Application Component Performance Criteria	Description
Cached Results Available	Signals whether the execution results of the application component are cached
Lowest Average Execution Time	The time the application component takes to run on each application server
Most Recently Executed	The application server that most recently ran the application component
Fewest Executions	The number of times the application component has run on each application server
Application Component Response Time	Average response time from a specific application server for the application component

Fig. 9

Carlson, Figures 8 and 9.

Figure 8 describes the server load criteria. The server load criteria include CPU load, disk input/output, memory thrash, number of requests queued, and server response time. Figure 9 describes the application component performance criteria. The application component performance criteria include cached results available, lowest average execution time. Most recently executed, fewest executions, and application component response time. However, *Carlson* does not teach "increasing a weight value associated with the application instance or the one or more other application instances based on which of the application instance and the one or more other application instances processes the least significant transactions" as recited in claim 21.

Additionally, *Carlson*'s figures 8 and 9 do not teach "increasing a weight value associated with the application instance or the one or more other application instances based on which of the application instance and the one or more other application instances passes transactions on to higher performing computing systems" as recited in new claim 22. *Carlson* is devoid of any such teaching.

Jindal also does not teach the recited features of new claims 21 and 22. *Jindal* teaches a method for load balancing requests for an application among a plurality of instances of the application operating on a plurality of servers. The relevant portion of *Jindal* cited by the Examiner is as follows:

> Other exemplary policies reflect preferences for the least-loaded instance of the application or the instance having the fastest response time. The least-loaded instance may be that which has the fewest connected clients and/or the fewest pending client requests. In another policy, where the closest instance of the application is favored, the preferred server may be the server that can be reached in the fewest network hops or connections. Another illustrative policy favors the server and/or the instance with the greatest throughput (e.g., the highest number of client requests satisfied in a given time period).

Jindal, column 2, lines 57-67.

The above cited portion of *Jindal* describes three load balancing policies. The first policy reflects preferences for the least-loaded instance of the application or the instance having the fastest response time. The least-loaded instance may be that which has the fewest connected clients and/or the fewest pending client requests. The second policy favors the closest instance of the application. The third policy favors the server and/or the instance with the greatest throughput.

Jindal does not teach a load balancing policy that favors application instances that processes the least significant transactions as recited in new claim 21. Additionally, *Jindal* does not teach a load balancing policy that favors application instances that passes transactions on to higher performing computing systems as recited in new claim 22. Therefore, the proposed combination of *Carlson* and *Jindal* when considered as a whole does not teach or suggest all of the features of the new claims.

III. **Conclusion**

The subject application is patentable over the cited references and should now be in condition for allowance. The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: October 12, 2007

Respectfully submitted,

/Gerald H. Glanzman/
Gerald H. Glanzman
Reg. No. 25,035
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorney for Applicants